



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/700,995	11/04/2003	Christopher Hahn	5486-0196PUS1	5612
67321	7590	12/12/2007	EXAMINER	
BIRCH, STEWART, KOLASCH & BIRCH, LLP			VU; TUAN A	
8110 GATEHOUSE ROAD				
SUITE 100 EAST			ART UNIT	PAPER NUMBER
FALLS CHURCH, VA 22040-0747			2193	
			MAIL DATE	DELIVERY MODE
			12/12/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)
	10/700,995	HAHN ET AL.
	Examiner	Art Unit
	Tuan A. Vu	2193

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --
Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) Responsive to communication(s) filed on 10/23/07.
- 2a) This action is FINAL. 2b) This action is non-final.
- 3) Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) Claim(s) 1-19 and 22 is/are pending in the application.
 - 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) Claim(s) _____ is/are allowed.
- 6) Claim(s) 1-19 and 22 is/are rejected.
- 7) Claim(s) _____ is/are objected to.
- 8) Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) The specification is objected to by the Examiner.
- 10) The drawing(s) filed on _____ is/are: a) accepted or b) objected to by the Examiner.
 Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
 Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
 - a) All b) Some * c) None of:
 1. Certified copies of the priority documents have been received.
 2. Certified copies of the priority documents have been received in Application No. _____.
 3. Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892)	4) <input type="checkbox"/> Interview Summary (PTO-413)
2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948)	Paper No(s)/Mail Date: _____
3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08)	5) <input type="checkbox"/> Notice of Informal Patent Application
Paper No(s)/Mail Date: _____	6) <input type="checkbox"/> Other: _____

DETAILED ACTION

1. This action is responsive to the Applicant's response filed 10/23/07.

As indicated in Applicant's response, claims 1, 6-7, 17 have been amended, and claims 20-21 canceled, and claim 22 added. Claims 1-19, 22 are pending in the office action.

Claim Rejections - 35 USC § 112

2. The following is a quotation of the first paragraph of 35 U.S.C. 112:

The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

3. Claims 1-5 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. The claim(s) contains subject matter which was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed invention.

Specifically, claim 1 recites instructions to operate on a data structure identifying parameter value combinations; and the data structure comprising a second section, **when instructed, extracts** a first set of parameter values **and lists** the ... parameter values in an order ... same order as the corresponding ... parameter order; a third section, **when instructed, extracts ... and lists** the second ... parameter values ... same order ... parameter order ...'. The data structure described in the Specifications (e.g. pg. 6) is perceived as either a spreadsheet or the markup file as illustrated in Figure 3. For one of ordinary skill in the art, data structure containing data listed in some order or position as claimed amounts to descriptive component or storage entity (e.g. Specs, Fig. 3); thus, the recited *second section* and *third section* comprising a

Art Unit: 2193

data structure (being either a spreadsheet or the XML file portions) cannot be construed as an functional entity when instructed, would extracts and lists. The inventor is deemed as not possessing the section in the data structure that would extracts and lists when instructed as set forth above. The above newly added subject matter would have no patentable weight and will be treated as a broadest possibility in the computer medium to perform '*extracts ... lists*' operation over the parameter combinations of the recited data structure. Removal of the un-supported claimed features is required.

Claims 2-5 for not remedying to the lack of support of the above new subject matter will also be rejected as failing to comply a proper written description of the claimed invention.

4. The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

5. Claims 1-5 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. Specifically, claim 1 recites 'second section ... extracts ... and lists ...'; 'third section ... extracts ... and lists...'. In light of the lack of written description from scanning the Disclosure to convey how a section in a data structure can extracts and lists, one of ordinary skill in the art would not be apprised of a clear context by which parameters can be listed, nor can the Disclosure enables one skill in the art to make use of the invention in terms of the (recited) listing (of parameters in some order from an extracting step) based from the above second and third section of a data structure. The extracting and listing step will be treated with very little patentable weight, and the data structure will be construed as a file with parameters listing, the

extracting and listing step treated with broadest interpretation allowable based on the lack of enabling Description from the Specifications as set forth in the § 112, 1st paragraph.

Claim Rejections - 35 USC § 101

6. 35 U.S.C. 101 reads as follows:

Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

The Federal Circuit has recently applied the practical application test in determining whether the claimed subject matter is statutory under 35 U.S.C. § 101. The practical application test requires that a “useful, concrete, and tangible result” be accomplished. An “abstract idea” when practically applied is eligible for a patent. As a consequence, an invention, which is eligible for patenting under 35 U.S.C. § 101, is in the “useful arts” when it is a machine, manufacture, process or composition of matter, which produces a concrete, tangible, and useful result. The test for practical application is thus to determine whether the claimed invention produces a “useful, concrete and tangible result”.

The current focus of the Patent Office in regard to statutory inventions under 35 U.S.C. § 101 for method claims and claims that recite a judicial exception (software) is that the claimed invention recite a practical application. Practical application can be provided by a physical transformation or a useful, concrete and tangible result. The following link on the World Wide Web is for the United States Patent And Trademark Office (USPTO) policy on 35 U.S.C. §101. <http://www.uspto.gov/web/offices/pac/dapp/opla/preognnotice/guidelines101_20051026.pdf>

7. Claims 1-5 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

Specifically, claim 1 recites ‘data structure’ comprising ‘a second section … extracts … and lists … in the parameter order’ and a third section … extracts … and lists … in the parameter order’. As identified from the USC 112 Rejection, the steps of extracting and listing by second and third section of the data structure (e.g. data structure: described in the Specs as a file or XML content) are deemed not provided with proper written description, hence would not amount to any action being readily and concretely realizable or remotely achieved. Any achieving being done in terms of (test) parameters being listed in some order therefore cannot

Art Unit: 2193

hold true; that is, any possible realization of any tangible result (e.g. list established tangibly in some form) from the claim as a whole remains impossible. The claim amounts to describing a computer medium having data structure therein, without any USC § 112 compliant step actions to put functionality of the medium instructions (software functionality) into a realization such as to yield a tangible and practical useful result. The claimed invention for not providing teaching as to fulfill the 3 prongs (concrete, tangible, and useful) of an statutory application result according to the above USC 101 Practical Application requirement, is rejected for a mere non-statutory subject matter.

Claims 2-5, for failing to remedy to the lack of teaching leading to realization of a useful and tangible result, are also rejected for a non-practical abstract idea.

Claim Rejections - 35 USC § 103

8. The following is a quotation of 35 U.S.C. §103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

9. Claims 1-5, 22 are rejected under 35 U.S.C. 103(a) as being unpatentable over Barhs et al. USPubN: 2003/0097650 (hereinafter Bahrs); in view of Bischof et al, USPubN: 2004/0041827 (hereinafter Bischof)

As per claim 1, Bahrs discloses a computer-readable medium having stored thereon computer executable instructions to operate on a data structure identifying parameter value

combinations the instructions when executed causes a computer system for use to test a software module, the data structure comprising:

a first section that includes a set of parameters listed in a parameter order for testing the software module (e.g. para 0051 – pg. 4; configuration file - para 0052 – pg. 5);

a second section, when instructed, extracts a first set of parameter values and lists the first set of parameter values, wherein the first set of parameter values is identified with a first test case for testing the software module (e.g. para 0038 – pg. 3 – Note: loading configuration files in a harness application from input forms as XML – see Fig. 5, para 0035-0036 – reads on first test case based on list of XML parameters extracted and listed in first or second test case formed via test mediator); and

a third section, when instructed, extracts a second set of parameter values and lists the second set of parameter values, (e.g. para 0038, pg. 3, Fig. 5; para 0035-0036) wherein the second set of parameter values is identified with a second test case for testing the software module (e.g. Fig. 7; multiple times – para 0042, pg. 4; multiple iterations – para 0045; specific data results ... passed to test component 404 - para 0035,pg. 3; steps 410, 412, 408 – Fig. 4).

Bahrs does not explicitly disclose that the first set and second set of extracted parameter values are listed in an order such that each value in said first set of parameter values is positioned in the same order as their respective and corresponding parameter listed in the first section parameter order. The markup language for creating tagged components or variables and associating tagged type or value thereof to those variables was known concept at the time the Invention was made. Based on sections of the markup configuration file by Bahrs (see para 0069, pg. 6) parallelism in defining of XML tags for named variables and their respective

value/type, as conveyed in Bahrs (e.g. <! --The duration of time the testing framework should be executing -- > , <! – specified amount of time -- > testDuration = “30000” … <!—The flag that indicates if this execution is to be threaded -- > , isThreaded = “True”) there is strong indication that a same order exist for the parameters and their corresponding values when a plurality of test cases (<TestCases> pg. 7) are created based on sections of the configuration file. Bischof, in a testing framework using markup script similar to the XML configuration file by Bahrs (see para 0041-0043, pg. 5-6) discloses named variables specified within a tag section having the same layout order as the corresponding value thereof in its associated tag section in an order of layout; see Bischof, pg. 6 – listing 1:

```
<Name> sendVKey </Name> and <Parameter type = “string”>0</Parameter> ;  
<Name> RS38M-PROGRAMM</Name> and <Type=GuiCTextField</Type> ;  
<Name> caretPosition</Name> and <Value type = “string”>18</Value> ;  
<Name> wnd[0]</Name> and <Value type=GuiMainWindow</Type> ); hence Bischof has disclosed extracted parameter values such that each value in said second set of parameter values (each of the parameter value) is positioned in the same layout order as the corresponding parameter has been listed in the source parameter tag section, i.e. the first section of test parameters. Based on the more test cases being derived from a plurality of XML configuration files, it would have been obvious for one skill in the art at the time the invention was made to implement the extraction of parameters in Bahrs XML file such that extracted parameters are listed in an order such that each value in said first set of parameter values is positioned in the same order as the corresponding parameter listed in the parameter order; and associating these extracted parameters and values for one first among more test cases. One would be motivated to
```

do so because this would enable associating first set of extracted values and second extracted values in the very order (as by Bischof as set forth above) with respect to their pertinent XML parameter order; and thereby implement test case input into Bahrs' harness application based on said parameter/values layout in order, as to render more efficient the parsing process effectuated by Bahrs' Test mediator (see para 0038-0045) handling of plurality of test cases.

As per claims 2-3, Bahrs discloses wherein the testing parameters are marked up with a markup language; wherein the markup language comprises the extensible markup language (see pg. 6-7).

As per claims 4-5, Bahrs does not explicitly disclose wherein the first section, second section and third section are elements of a table comprises additional sections that include sets of parameter. But based on Bahrs' listing of parameter values visualized with respect to their Attribute, name, Description (see Fig. 8-9, 11) it would have been obvious for one skill in the art at the time the invention was made to implement the XML configuration file by Bahrs so that the data structure first, second and third section in said XML form are derived from reading specifications from a table representing the needs of the above software development, test and analysis (see Fig. 3), i.e. test specification/description by Bahrs; because analysis stage use of table structure as in Bahrs approach would (see Fig. 3) be able to map values to their parameters prior to these representation be implemented in the configuration file (see Fig .7; Fig 12) based on the architecture and needs analysis.

As per claim 22, Bahrs discloses a computer-readable medium having stored thereon computer executable program for testing a software module, the computer program when executed causes a computer system to execute steps of:

extracting a set of parameters listed in a parameter order (e.g. Fig. 12) for testing the software module;

extracting a first set of parameter values and listing the first set of parameter values, wherein the first set of parameter values is identified with a first test case for testing the software module (para 0038 – pg. 3 – Note: loading configuration files in a harness application from input forms as XML – see Fig. 5, para 0035-0036 – reads on first test case based on list of XML parameters extracted and listed in first or second test case formed via test mediator);

testing the software module based on the first test case (e.g. Fig. 7);

extracting a second set of parameter values and listing the second set of parameter values wherein the second set of parameter values is identified with a second test case for testing the software module (para 0038, pg. 3, Fig. 5 – Note: Note: loading configuration files in a harness application from input forms as XML – see Fig. 5, para 0035-0036 – reads on first test case or second test case based on list of XML parameters extracted and listed in first or second test case formed via test mediator); and testing the software module based on the second test case (e.g. Fig. 7; multiple times – para 0042, pg. 4; multiple iterations – para 0045; specific data results ... passed to test component 404 - para 0035,pg. 3; steps 410, 412, 408 – Fig. 4).

But Bahrs does not explicitly disclose that the first set and second set of extracted parameter values are listed in an order such that each value in said first set of parameter values is positioned in the same order (parameter order) as their respective and corresponding parameters as they are listed in the extracted list of parameter.

But this limitation has been rendered obvious based on the rationale of claim 1 using Bischof.

10. Claims 6-19 are rejected under 35 U.S.C. 103(a) as being unpatentable over Barhs et al.

USPubN: 2003/0097650 (hereinafter Bahrs)

As per claim 6, Bahrs discloses a method of processing testing data for testing a software module, the method comprising:

extracting parameter value combinations from a data file formatted with a markup language to implement data associated with a first test case (e.g. para 0051 – pg. 4; configuration file - para 0052 – pg. 5);

transmitting the parameter value combinations to a software module test engine, wherein the parameter value combinations are identified with the first test case (see Fig. 5, 12; para 0057-0058, pg. 5); and

testing the software module with the parameter value combinations based on the first test case (see Fig. 7) ;

generating a first test result based on the first test case (Fig. 4);

generating a second test result based on the second test case (e.g. specific data results ... passed to test component 404 - para 0035,pg. 3; steps 410, 412, 408 – Fig. 4).

But Bahrs does not explicitly disclose (i) markup language to implement data of the *external table* associated with first test case; nor does Bahrs disclose (ii) changing the data file to implement data of the external table associated with a second test case for testing the software module, wherein the parameter value combinations are identified with the second test case.

Based on the rationale as set forth in claims 4-5, the limitation (i) as to implement the markup file from an *external table* has been addressed as obvious.

Bahrs integration testing (see Fig 6); and analysis of results to implement more testing applied to diverse parts of the project software has been conveyed in Figure 4, including reuse based on modifications of attributes (see para 0052, pg. 5) for instance of test, reusability in terms of test component or test cases (see para 0059, pg. 5; *only changing configuration information* - para 0065, pg. 6) expressed via use of the extensible language configuration (see para 0071, pg. 8). Based on the reusability aspect of XML reconfiguration of test cases by Bahrs, it would have been obvious for one skill in the art at the time the invention was made to effectuate analysis of results as by Bahrs so that the requirement needed for the components of a target software (refer to claims 4-5) can be modified as suggested above; that is, table data specifying the data for implementing test and the extensible markup file being subsequently modified as a results of matching expected values against test results (see Bahrs: Fig. 4, 13) and associate additional changes to the table and the configuration parameters (i.e. *wherein the parameter value combinations are identified with the second test case*) for effectuating additional test based (i.e. *changing the data file to implement data of the external table associated with a second test case for testing the software module*) based on the modified parameters. One of ordinary skill in the art would do so because based on the endeavor by Bahrs to derive further specifications from results analysis and iterate more test instances therefrom (multiple times – para 0042, pg. 4; multiple iterations – para 0045; specific data results ... passed to test component 404 - para 0035,pg. 3; steps 410, 412, 408 – Fig. 4); i.e. such modifying enabling more instances of test cases which would expedite dynamic and recursive testing of every part of the whole project based on the results, and thus alleviating time and cost that would

otherwise be consuming development resources (see Bahrs: Background, para 0071, pg. 8; para 0035, pg. 3; *as much reusable ... as feasible* – para 0036, pg 3).

As per claim 7, Bahrs discloses that the data file comprises a table containing a plurality of test cases and each test case comprises a set of parameter value combinations (refer to rationale as set forth in claims 4-5)

As per claims 8-9, Bahrs (in view of the rationale in claims 4-5) discloses wherein extracting comprises extracting the plurality of test cases from the data file including creating an object from a test case parameter value combination (see Fig. 5, 12; para 0057-0058, pg. 5)

As per claims 10, and 12, Bahrs discloses changing the format (e.g. can develop also alternate behavior – para 0040 , pg. 3; *as much reusable ... as feasible* – para 0036, pg 3) of the parameter value combinations before transmitting, including validating the parameter value combinations by comparing the parameter value combinations to a set of rules (e.g. *ItestCase ... contract* – para 0041; *guarantees that all test case will ... implementation of a method, exception handling* – para 0041, pg. 4).

As per claim 11, Bahrs does not explicitly disclose receiving a table of parameter value combinations at a spreadsheet application; and converting the table to the data file with a spreadsheet plug-in.

But based on table layout (refer to claims 4-5) as suggested via Bahrs' test requirement and specification from which to formulate XML file statements in terms of markup data configuration, the setting of parameters (see Fig. 8-9, 11) as suggested via use of external table to implement XML configuration test file has been addressed above. At the time that the

invention was made, the Spreadsheet technology having its internal macro to facilitated dynamic update of data cells was a well-known concept, and one skill in the art would be motivated to implement such spreadsheet as above. One would be motivated to do so because based on Bahrs' update a instance of parameter combination using deletion option (see para 0052, pg. 5) for reusing test components in the process of finalizing parameter collection for more testing, the use of spreadsheet with its macro editing functions would enhance such dynamic update of parameter translation files to support Bahrs's markup file (Note: the very nature of XML is that they are extensible); that is, these can be fine tuned with the Spreadsheet macro options Bahrs's endeavor to modify parametes file based on dynamic test results (see claim 6) and this is consistent with flexible aspect by Bahrs by which test coverage can be modified (refer to claim 6) hence enhancing it by continual updating of requirements via feedback from Bahrs's test suites evaluation and testing tool.

As per claim 13, Bahrs discloses wherein parameter value combinations are validated on demand prior to (e.g. para 0037 – pg 3; Fig. 5; by a developer – para 0041, pg. 4; can develop also alternate behavior – para 0040 , pg. 3 – Note: developer's implementing of parameters and invoking of mediator interface to contract the class method with specific constructor guarantees and/or exception handling reads on on-demand invoking of code to validate Java constructs based on configuration file parameter extraction).

As per claim 15, Bahrs discloses a medium (see computer – Fig. 1; para 0022, pg. 2) for performing the steps recited in claim 11.

As per claims 14, 16, Bahrs discloses a medium (see computer – Fig. 1, para 0022, pg. 2)

As per claim 17, Bahrs discloses a computer-readable medium containing computer-executable components comprising:

an import component that extracts parameter value combinations from a data file formatted with a marked up language to implement data of an external table associated with a first test case (para 0051 – pg. 4; configuration file - para 0052 – pg. 5);

a test object creation module that creates an object to test a software module with the parameter value combinations associated with the first test case (see Fig. 7);

wherein the import component is configured to extract parameter value combinations from the data file to implement data of the external table associated with a second test case for testing the software module (Note: para 0038, pg. 3, Fig. 5 – Note: Note: loading configuration files in a harness application from input forms as XML – see Fig. 5, para 0035-0036 – reads on first test case or second test case based on list of XML parameters extracted and listed in first or second test case formed via test mediator).

As per claim 18, Bahrs discloses the markup language comprises the extensible markup language (re claim 3).

As per claim 19, Bahrs discloses wherein the import module validates the parameter value combinations (refer to claims 12-13).

Response to Arguments

11. Applicant's arguments filed 10/23/07 have been fully considered but they are not moot in light of the new grounds of rejection. Following are the Examiner's observation in regard thereto.

Conclusion

Art Unit: 2193

12. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Tuan A Vu whose telephone number is (571) 272-3735. The examiner can normally be reached on 8AM-4:30PM/Mon-Fri.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Meng-Ai An can be reached on (571)272-3756.

The fax phone number for the organization where this application or proceeding is assigned is (571) 273-3735 (for non-official correspondence - please consult Examiner before using) or 571-273-8300 (for official correspondence) or redirected to customer service at 571-272-3609.

Any inquiry of a general nature or relating to the status of this application should be directed to the TC 2100 Group receptionist: 571-272-2100.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).



Tuan A Vu
Patent Examiner,
Art Unit 2193
December 09, 2007